

Standard Form 298 (Rev. 2-89)
Prescribed by GSA FPMR (41 CFR) 101-11.6
298-102

Covering, Packing and Logical Inference:
Final Technical Report

E. BALAS
G. CORNUÉJOLS
J. N. HOOKER
Graduate School of Industrial Administration
Carnegie Mellon University
Pittsburgh, PA 15213 USA

AFOSR Grant No. 91-0287

October 1993

Accession For	
NTIS CRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution /	
Availability Codes	
Dist	Available for Special
A-1	

DTIC QUALITY INSPECTED 3

93-31288
■■■■■■■■■■

93 12 27 054

Abstract

We report progress in the following areas. (Some of these results reflect joint work with others.)

Set covering problems. We developed a new heuristic based on a subgradient approach to solving the Lagrangean dual. As a stand-alone heuristic it compares favorably with others. When used to replace the conventional subgradient procedure in an exact algorithm based on branch-and-bound, it improves performance drastically.

Problems soluble by integer programming We showed that logical inference problems defined by a "balanced" matrix are soluble both by linear programming and by a nonnumeric algorithm similar to unit resolution. We showed how to recognize balanced 0,1 matrices in polynomial time, using graph decomposition, and made progress toward the recognition of 0, ± 1 balanced matrices. We contributed to the recognition of LP-soluble set covering problems by expanding the list of known minimal nonideal submatrices. We proved that the resolution method of theorem-proving allows one to check whether a satisfiability problem can be solved as a linear programming problem by checking the same for all of its set covering subproblems.

Inference in propositional logic. We systematically tested inference algorithms for propositional logic and identified which ones seem to perform well on easy, medium-difficulty, and hard problems, and to some extent why they do. We developed and tested a logic circuit verification algorithm based on Benders decomposition that is better than the state of the art on certain classes of circuits and worse on others. We showed how to solve the problem, "what are *all* the implications of a rule base with respect to a given question," as a projection problem—completely in the Horn case, and partially in the non-Horn case. We showed how to generate good ("tight") MILP formulations of logical rules (in particular, cardinality rules). We moved closer to the goal of an efficient solver for non-Horn logic programming by extending partial instantiation methods to full first-order logic with function symbols. We applied logical inference methods to chemical engineering design problems.

Inductive, uncertainty and belief logics. We unified several uncertainty and belief logics by showing that they can be solved by a single linear programming algorithm with a different column generation subroutine for each logic. We showed how to

combine statistical and boolean methods to obtain a regression-based method for deriving rules for an expert system based on past expert behavior.

Contents

1	Introduction	5
2	Set Covering Problems	5
3	Recognizing Packing, Covering and Logical Inference Problems with an Integral Underlying Polyhedron	7
3.1	A Class of Logic Problems Solvable by Linear Programming .	7
3.2	Recognizing balanced 0, 1 matrices	7
3.3	Ideal Matrices	8
3.4	Resolution and Integrality	9
4	Inference in Propositional Logic	9
4.1	Computational Testing of Satisfiability Algorithms	9
4.2	Logic Circuit Testing	10
4.3	Inference as Projection	10
4.4	Formulation of Logic Constraints.	10
4.5	Logic Programming	10
4.6	Logic-Based Methods for Design	11
5	Inference Methods for Belief Systems	11
5.1	A Linear Programming Framework for Combining Evidence. .	11
5.2	Obtaining Rules for Expert Systems	11

1 Introduction

This is the final technical report for research partially supported by the AFOSR grant, "Covering, Packing and Logical Inference," which took effect on 1 July 1991. This report follows the same outline as the proposal.

The papers written in connection with this grant and its predecessor, "Mathematical Programming and Logical Inference," are marked with an asterisk in the bibliography. Papers written under the present grant are marked with a double asterisk.

2 Set Covering Problems

In a recent joint paper with Maria C. Carrera, "A Dynamic Subgradient-Based Branch and Bound Procedure for Set Covering" [5], we report on the development, implementation and computational testing of a class of algorithms for the set covering problem that has shown excellent computational performance on a large battery of test problems.

We address the set covering problem

$$(SC) \quad \min\{cx \mid Ax \geq 1, x \in \{0, 1\}^n\},$$

where A is an $m \times n$ matrix of 0's and 1's, c is an integer n -vector, and 1 is the m -vector of 1's.

Our approach belongs to a family of branch and bound procedures for (SC) whose main characteristic is that at every node of the search tree, instead of using the simplex method to solve the linear programming relaxation of the given subproblem, it uses some combination of primal and dual heuristics with subgradient optimization applied to a Lagrangean dual, possibly incorporating cutting planes, to generate upper and lower bounds on the objective function value.

Various elements of this approach have been around for about 15 years. Etcheberry [25] seems to have been first to use subgradient optimization instead of the simplex method. Balas and Ho [6] tested various Lagrangean relaxations, some of them including cutting planes, and introduced several primal and dual heuristics combined with variable fixing techniques, as well as some new branching rules. Their algorithm and its computational performance served as a benchmark for subsequent developments in the 80's. Hall and Hochbaum [27] extended the approach to more general covering problems (right hand side greater than one). Vasko and Wilson [49] introduced

an efficient randomized version of the greedy heuristic. Carrera [15] performed a thorough computational study of primal heuristics, including the randomized greedy and reduced cost heuristics. Beasley [10] implemented a branch and bound algorithm using the Balas-Ho framework, but solving the linear programming subproblems by the simplex method and using some new variable fixing rules. In a later paper, Beasley [11] introduced a Lagrangean heuristic that generates a cover after every subgradient iteration. Fisher and Kedia [26] use an efficient dual heuristic as the main ingredient of their approach. For some recent work on set covering algorithms based on other approaches, see Harche and Thompson [29] and Nobili and Sassano [45].

The centerpiece of the approach discussed in this paper is an integrated lower bounding/upper bounding procedure that we call dynamic subgradient optimization (DYNSGRAD) and that we apply to a Lagrangean dual problem at every node of the search tree. This procedure intertwines the iterations of subgradient optimization, a lower bounding procedure, with applications of the reduced cost heuristic (RCH), a primal, i.e. upper bounding procedure (which however, as a byproduct, also improves the lower bound), followed by variable fixing applied in a recursive fashion. Whenever RCH improves the upper bound, the current dual vector (set of Lagrange multipliers) is changed, along with the upper and lower bounds; and whenever some variable is fixed at 1, the constraint set of the Lagrangean problem itself changes, and the parameters of the subgradient procedure are adjusted: hence the qualifier "dynamic" in the name of the procedure.

Other new features of our algorithm include some primal and dual heuristics, a recursive variable fixing procedure, and new branching rules.

Our procedures have been extensively tested, both on randomly generated and on real world problems. As a stand-alone heuristic, the dynamic subgradient procedure compares favorably with all other heuristics known to us in terms of the quality of solutions obtainable with a certain computational effort. As a new ingredient of our branch and bound procedure, it has drastically improved the performance of the latter as compared to its older version which uses at every node the standard subgradient optimization procedure. It also compares favorably with other branch and bound procedures.

3 Recognizing Packing, Covering and Logical Inference Problems with an Integral Underlying Polyhedron

3.1 A Class of Logic Problems Solvable by Linear Programming

In propositional logic, several problems such as satisfiability, MAXSAT and logical inference, can be formulated as integer programs. For example, given a set S of clauses and a weight vector w whose components are indexed by the clauses in S , the *weighted maximum satisfiability problem (MAXSAT)* consists in finding a truth assignment that maximizes the total weight of the satisfied clauses. MAXSAT can be formulated as the integer program

$$\begin{aligned} \text{Min } & \sum_{i=1}^m w_i s_i \\ & Ax + s \geq 1 - n(A) \\ & x \in \{0, 1\}^n, s \in \{0, 1\}^m \end{aligned}$$

where A is a $0, \pm 1$ matrix.

The above three problems are NP-hard in general but SAT and logical inference can be solved efficiently for Horn clauses, clauses with at most two literals and several related classes [17],[48]. MAXSAT remains NP-hard for Horn clauses with at most two literals.

In [19], we show that the above three problems can be solved in polynomial time, as linear programs, whenever the matrix A associated with S is a balanced $0, \pm 1$ matrix as defined by Truemper [47], see also [46]. A $0, \pm 1$ matrix A is *balanced* if, in every submatrix of A with exactly two nonzero entries per row and per column, the sum of the entries is a multiple of 4.

The key to obtaining this result is the following theorem. If A is a balanced $0, \pm 1$, then $R(A) = \{(x, s) \in \mathcal{R}^{n+m} : Ax + s \geq 1 - n(A), 0 \leq x, s \leq q1\}$ is an integer polytope.

3.2 Recognizing balanced $0, 1$ matrices

Our approach to recognizing balanced matrices uses graph decomposition. To a $0, 1$ matrix A we associate a bipartite graph $G(A)$ as follows: the node set of $G(A)$ is partitioned into V^r and V^c which represent the row set and the column set of A , and $G(A)$ has an edge connecting node i and node j if and only if $a_{ij} = 1$. A $0, 1$ matrix is balanced if and only if the length of every chordless cycle in the associated bipartite graph is a multiple of 4.

A decomposition theorem is established in [20] Parts II-VI and used in Part VII to recognize in polynomial time whether a 0,1 matrix is balanced. A 2-join in a bipartite graph G is a set of edges $E_1 \cup E_2$ such that, for $i = 1, 2$, E_i induces a complete bipartite subgraph G_i of G , the graphs G_1 and G_2 have disjoint node sets and $G \setminus (E_1 \cup E_2)$ is disconnected. A *double star cutset* in a bipartite graph G is a node set S such that $G \setminus S$ is disconnected and there exist two adjacent nodes u, v with the property that every node of S is adjacent to u or v . Restricted balanced matrices are those where every cycle of $G(A)$ has length congruent to 0 modulo 4. Yannakakis [50] showed how to recognize them in linear time.

Our decomposition theorem states that, if a balanced bipartite graph is not restricted balanced, then it contains a 2-join or a double star cutset.

Double star cutsets and 2-joins can be used recursively to decompose a given bipartite graph G into elementary blocks that contain neither. In [20], it is shown how to perform this decomposition so that only a polynomial number of elementary blocks are created and G is balanced if and only if all the elementary blocks are restricted balanced. A polynomial algorithm for recognizing balanced 0,1 matrices follows.

The decomposition theorem for balanced 0,1 matrices has been recently extended to balanced $0, \pm 1$ matrices in [21] but a new decomposition is required as well as a new elementary block which is not restricted balanced. The algorithmic issues are currently under investigation.

3.3 Ideal Matrices

A 0,1 matrix M is *ideal* if all vertices of the set covering polyhedron $\{x : Mx \geq 1, x \geq 0\}$ have only 0,1 components. The 0,1 matrix A is said to be minimally nonideal if the linear system $Ax \geq 1$ has no redundant constraints, the polyhedron $P(A)$ has a fractional extreme point but all the polyhedra obtained by intersecting $P(A)$ with one of the hyperplanes $x_j = 0$ or $x_j = 1$ for $j = 1, \dots, n$ are integral. Minimally nonideal matrices are important since they represent the fundamental violators of idealness, i.e. every nonideal matrix must contain a minimally nonideal matrix. The study of minimally nonideal matrices is the counterpart for the set covering problem of the study of minimally imperfect matrices for the set packing problem. In [23], we expand the list of known minor minimal nonideal matrices by several hundred. Many of these examples are obtained polyhedrally, by constructing new minimally nonideal matrices from old ones. We present a conjecture that might be viewed as the counterpart for ideal matrices

of Berge's Strong Perfect Graph Conjecture. We provide evidence for the conjecture by completely characterizing all minimally nonideal circulants.

3.4 Resolution and Integrality

J. Hooker identified another class of inference problems that can be solved by linear programming [39]. If resolution is applied to a satisfiability problem, then it describes an integral polytope (and is therefore soluble by LP) if and only if its set covering subproblems do. A corollary of this work is that the facets of the ssatisfiability polytope consists of facets of various set covering polytopes plus prime implications obtained by resolution.

4 Inference in Propositional Logic

4.1 Computational Testing of Satisfiability Algorithms

F. Harche, J. Hooker and G. Thompson did a computational study of several satisfiability algorithms based on tree search, including versions of the Davis-Putnam-Loveland, Jeroslow-Wang, Horn relaxation, branch-and-cut, and column subtraction algorithms [28]. All testing was done on the same machine using as many common subroutines as possible. The problems were tested on a large set of benchmark problems collected by F. J. Radermacher. It was found that methods that employ weak relaxations at the search tree nodes, such as the Horn relaxation method, are much more effective for the easier problems and much less effective for the harder problems. The column subtraction method was the most robust, as it was the only method capable of solving all the problems in the set. The Jeroslow-Wang, branch-and-cut and one version of the Davis-Putnam-Loveland are fastest on problems of intermediate difficulty.

Based partly on this experience, J. Hooker developed a fast algorithm for the incremental satisfiability problem [36]. This is the problem of checking whether a satisfiable set of propositions remains satisfiable when another proposition is added. It is a key subproblem for logic circuit verification [42] and partial instantiation methods for first order predicate logic [35]. Computational study showed that in most cases, using the incremental algorithm is an order of magnitude faster than re-solving the problem from scratch.

4.2 Logic Circuit Testing

J. Hooker and H. Yan developed and tested a new algorithm for logic circuit verification, which is based on Benders decomposition. This is a problem on which Japanese and American industry has worked intensively for several years. They compared the performance of their prototype algorithm with the most recent state-of-the-art code (KBDD).

The Benders code ran more slowly than KBDD on most of the standard benchmark problems in the literature. Hooker and Yan attributed this to the large number of exclusive-OR gates in these problems. They generated random problems without exclusive-OR gates and found that the Benders approach is an order of magnitude faster than KBDD on almost all of them. They conclude that the Benders approach is very effective on certain classes of problems and ineffective on others, and its strengths are somewhat complementary to those of BDD algorithms.

4.3 Inference as Projection

The problem of deriving all inferences in propositional logic that answer a given question (i.e., that contain a given set of atomic propositions) is a type of projection problem. J. Hooker wrote a paper [37] that shows that solving a related polyhedral projection problem is equivalent to applying a form of unit resolution to the logical problem. This provides a practical algorithm for solving Horn projection problems and partially solving non-Horn problems.

4.4 Formulation of Logic Constraints.

J. Hooker and H. Yan wrote a paper on tight formulations of cardinality rules [43]. These results show how to get good representations of complex logical constraints, such as a rule base, in a mixed integer programming model.

4.5 Logic Programming

Current logic programming systems (PROLOG, PROLOG III etc.) provide inference methods that at best are complete only for datalog languages (universally quantified first-order Horn logic). J. Hooker and G. Rago [41] extended the partial instantiation approach of R. Jeroslow to accommodate function symbols, so that it becomes applicable to full non-Horn first-order

logic. To avoid the problem of undecidability, they introduced a device that checks whether a logic program has a model that involves a given maximum level of function nesting, which is decidable. Work to date lays the theoretical foundations.

4.6 Logic-Based Methods for Design

J. Hooker, H. Yan, I. Grossmann, and R. Raman wrote a paper [44] that shows how logic cuts can be used to speed the solution of MILP models for chemical process design problems.

5 Inference Methods for Belief Systems

5.1 A Linear Programming Framework for Combining Evidence.

Andersen and Hooker reviewed the AI literature on evidential reasoning and discovered that many uncertainty logics fit the same LP model. They show in a new paper [3] how to implement several logics in a single LP framework by "plugging in" different column generation subroutines for the different logics. In particular they formulated two practical logics that account for second order probabilities (i.e., the reliabilities of the sources of probability information).

J. Hooker also published a survey paper on mathematical programming methods for reasoning under uncertainty [38].

5.2 Obtaining Rules for Expert Systems

E. Boros, P. Hammer and J. Hooker showed how an approach analogous to statistical regression can be used to generate rules for an expert system, based on the past decisions of experts. It has the advantage that statistical tests can be used to decide whether the derived rules are statistically valid.

They also moved partway toward extending this work to the nonboolean case. In a second paper [13], they show how a network flow model can be used to derive the best-fitting (nonboolean) rules when they are required to be "monotone."

References

- [1] *K. A. Andersen and J. N. Hooker, Bayesian logic, *Decision Support Systems* 9 (1993).
- [2] *Andersen, K. A., and J. N. Hooker, Probabilistic Logic for Belief Nets, *Proceedings, 8th International Congress of Cybernetics and Systems*, New York City (June 1990) 245-251.
- [3] **Andersen, K. A., and J. N. Hooker, A linear programming framework for logics of uncertainty, *Proceedings of HICSS92* (Hawaii International Conference on Systems Sciences), Washington, IEEE Computer Society Press (1992) 533-542.
- [4] *Balas, E., On the convex hull of the union of certain polyhedra, *Operations Research Letters* 7 (1988) 279-283.
- [5] **Balas, E., and M. C. Carrera, A dynamic subgradient-based branch and bound procedure for set covering, Management Sciences Research Report No. MSRR 568(R), Graduate School of Industrial Administration, Carnegie Mellon University, Pittsburgh, PA 15213 USA, May 1992.
- [6] Balas, E., and A. Ho, Set covering algorithms using cutting planes, heuristics and subgradient optimization: A computational study. *Mathematical Programming* 12 (1980) 37-60.
- [7] *Balas, E., and S. M. Ng, On the set covering polytope: I. All the facets with coefficients in $\{0, 1, 2\}$, *Mathematical Programming* 43 (1989) 57-69.
- [8] *Balas, E., and S. M. Ng, On the set covering polytope: II. Lifting the facets with coefficients in $\{0, 1, 2\}$, *Mathematical programming* 45 (1989) 1-20.
- [9] *Balas, E., and J. Xue, Minimum weighted coloring of triangulated graphs, with application to weighted vertex packing in arbitrary graphs, to appear in *SIAM Journal on Computing*.
- [10] Beasley, J. E., An algorithm for set covering problems, *European Journal of Operational Research* 31 (1987) 85-93.

- [11] Beasley, J. E., A lagrangean heuristic for set covering problems, *Naval Research Logistics* **37** (1990), 151-164.
- [12] Beasley, J. E., and K. Jornsten, Enhancing an algorithm for set covering problems, *European Journal of Operational Research* **58** (1992) 293-300.
- [13] **Boros, E., P. L. Hammer and J. N. Hooker, Predicting cause-effect relationships from incomplete discrete observations, 1991, to appear in *SIAM Journal on Discrete Mathematics*.
- [14] **Boros, E., P. L. Hammer and J. N. Hooker, Boolean regression, 1991, submitted for publication.
- [15] Carrera, M. C., Computational study of set covering heuristics, Master's thesis, Department of Mathematics, Carnegie Mellon University, 1984.
- [16] *Chandru, V., and J. N. Hooker, Logical inference: A mathematical programming perspective, in S. T. Kumara, R. L. Kashyap, and A. L. Soyster, eds., *Artificial Intelligence: Manufacturing Theory and Practice*, IIE (1988) 97-120.
- [17] *Chandru, V., and J. N. Hooker, Extended Horn sets in propositional logic, *Journal of the ACM* **38** (1991) 203-221.
- [18] *Chandru, V., and J. N. Hooker, *Optimization Methods for Logical Inference*, to be published by Wiley.
- [19] **Conforti M., and G. Cornuéjols, A class of logic problems solvable by linear programming, submitted to *Journal of the ACM*.
- [20] Conforti M., and G. Cornuéjols, M. R. Rao, Decomposition of balanced 0,1 matrices, Parts I-VII, preprints, Carnegie Mellon University (1991).
- [21] Conforti M., and G. Cornuéjols, A. Kapoor, K. Vuskovic, Decomposition of balanced 0,+1,-1 matrices, Parts I-III, preprints, Carnegie Mellon University (1993).
- [22] **Cornuéjols G., and B. Novick, Ideal 0,1 matrices, to appear in *Journal of Combinatorial Theory B*.
- [23] *Cornuéjols, G., and A. Sassano, On the 0,1 facets of the set covering polytope, *Mathematical Programming* **43** (1989) 45-55.

- [24] Etcheberry, J., The set covering problem: A new implicit enumeration algorithm. *Operations Research* **25** (1977) 760-772.
- [25] Fisher, M. L., and P. Kedia, Optimal solution of set covering problems using dual heuristics, *Management Science* **36** (1990) 674-688.
- [26] Hall, N. G., and D. S. Hochbaum, A fast approximation algorithm for the multicovering problem. Technical Report, Ohio State University, 1985.
- [27] **Harche, F., J. N. Hooker and G. L. Thompson, A computational study of satisfiability algorithms for propositional logic, to appear in *ORSA Journal on Computing*.
- [28] Harche, F., and G. L. Thompson, The column subtraction algorithm: An exact method for solving weighted set covering problems, MSRR No. 548, GSIA, Carnegie Mellon University, 1989.
- [29] *Hooker, J. N., Resolution vs. cutting plane solution of inference problems: Some computational experience, *Operations Research Letters* **7** (1988) 1-7.
- [30] *Hooker, J. N., Generalized resolution and cutting planes, *Annals of Operations Research* **12** (1988) 217-239.
- [31] *Hooker, J. N., A quantitative approach to logical inference, *Decision Support Systems* **4** (1988) 45-69.
- [32] *Hooker, J. N., Input proofs and rank one cutting planes, *ORSA Journal on Computing* **1** (1989) 137-145.
- [33] **Hooker, J. N., New methods for inference in first-order logic, *Annals of Operations Research* **43** (1993).
- [34] *Hooker, J. N., Generalized resolution for 0-1 linear inequalities, *Annals of Mathematics and AI* **6** 271-286.
- [35] **Hooker, J. N., Solving the incremental satisfiability problem, *Journal of Logic Programming* **15** (1993) 177-186.
- [36] **Hooker, J. N., Logical inference and polyhedral projection, *Computer Science Logic Conference 1991, Lecture Notes in Computer Science* **626** (1992) 184-200.

- [37] **Hooker, J. N., Mathematical programming methods for reasoning under uncertainty, *Operations Research Proceedings 1991*, Deutsche Gesellschaft für Operations Research, Springer (1992) 23-34.
- [38] **Hooker, J. N., Resolution and the integrality of satisfiability problems, 1992, submitted for publication.
- [39] *Hooker, J. N. and C. Fedjki, Branch-and-cut solution of inference problems in propositional logic, *Annals of Mathematics and AI* 1 (1990) 123-140.
- [40] **Hooker, J. N., and G. Rago, Partial instantiation methods for logic programming, in preparation.
- [41] *Hooker, J. N., and H. Yan, Verifying logic circuits by Benders decomposition, 1991, submitted to *SIAM Journal on Computing*.
- [42] **Hooker, J. N., and H. Yan, Tight representation of logical constraints as cardinality rules, 1992, submitted to *Discrete Applied Mathematics*.
- [43] **Hooker, J. N., H. Yan, I. Grossmann, R. Raman, Logic cuts for processing networks with fixed charges, *Computers and Operations Research* 21 (1993).
- [44] Nobili, P., and A. Sassano, A Separation routine for the set covering polytope, in E. Balas, G. Cornuejols and R. Kannan (editors), *Integer Programming and Combinatorial Optimization*, Proceedings of the 2nd IPCO Conference, Carnegie Mellon University Press, 1992.
- [45] Schrijver A., *Theory of linear and integer programming*, Wiley, New York (1986).
- [46] Truemper K., On balanced matrices and Tutte's characterization of regular matroids, preprint (1978).
- [47] Truemper K., Polynomial theorem proving I. Central matrices, *Technical Report UTDCS 34-90* (1990).
- [48] Vasko, F. J., and G. R. Wilson, An efficient heuristic for large set covering problems, *Naval Research Logistics Quarterly* 31 (1984) 163-171.
- [49] Yannakakis M., On a class of totally unimodular matrices, *Mathematics of Operations Research* 10 (1985) 280-304.